WallTokens: Surface Tangibles for Vertical Displays

Emmanuel Courtoux, Caroline Appert, Olivier Chapuis

WallTokens: Surface Tangibles for Vertical Displays

Emmanuel Courtoux
Université Paris-Saclay
CNRS, Inria, LISN
Orsay, France

Caroline Appert
Université Paris-Saclay
CNRS, Inria, LISN
Orsay, France

Olivier Chapuis
Université Paris-Saclay
CNRS, Inria, LISN
Orsay, France

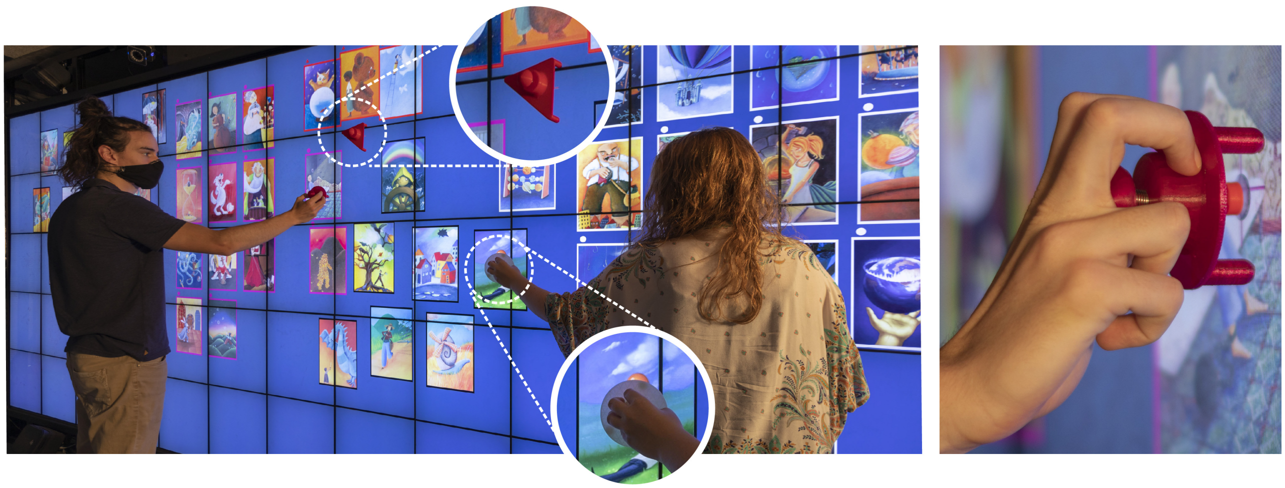

Figure 1: (Left) Two users interacting with *WallTokens* on a wall display. Each *WallToken* can be manipulated on the surface or attached to it. (Right) Close up of a user attaching a *WallToken* to the wall.

ABSTRACT

Tangibles can enrich interaction with digital surfaces. Among others, they support eyes-free control or increase awareness of other users' actions. Tangibles have been studied in combination with horizontal surfaces such as tabletops, but not with vertical screens such as *wall displays*. The obvious obstacle is gravity: tangibles cannot be placed on such surfaces without falling. We present *WallTokens*, easy-to-fabricate tangibles to interact with a vertical surface. A *WallToken* is a passive token whose footprint is recognized on a tactile surface. It is equipped with a *push-handle* that controls a suction cup. This makes it easy for users to switch between sliding the token or attaching it to the wall. We describe how to build such tokens and how to recognize them on a tactile surface. We report on a study showing the benefits of *WallTokens* for manipulating virtual objects over multi-touch gestures. This project is a step towards enabling tangible interaction in a wall display context.

CCS CONCEPTS

• **Human-centered computing** → **Haptic devices**; **Gestural input**; *Collaborative interaction*.

CHI '21, May 8–13, 2021, Yokohama, Japan

KEYWORDS

Tangible Interaction; Tokens; Wall-sized display; Fabrication

ACM Reference Format:
Emmanuel Courtoux, Caroline Appert, and Olivier Chapuis. 2021. WallTokens: Surface Tangibles for Vertical Displays. In *CHI Conference on Human Factors in Computing Systems (CHI '21), May 8–13, 2021, Yokohama, Japan.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3411764.3445404

1 INTRODUCTION

Large, ultra-high resolution displays (*wall displays*) make it possible for one or several users to interact with large volumes of data. Unlike horizontal screens such as tabletops, vertical displays make it possible for an audience to observe a scene from roughly the same perspective, which is particularly important when that scene has a preferred orientation – *e.g.*, text documents or maps. Users can also explore that scene at different levels of detail through physical navigation (stepping close to see the details, stepping back to gain an overview). However, designing efficient interaction techniques for large vertical displays is particularly challenging. Distant interaction can rely on mid-air gestures [36] or personal devices used as remote controllers [6, 21]. Close interaction can rely on touch gestures (*e.g.*, [30]). The latter have the advantage of leaving users' hands free of any controller, but they also lack precision and hardly scale to concurrent interactions from multiple users.

Surface tangibles can enrich interaction from close distance with a large surface. As opposed to tangibles that are tracked in the air (with *e.g.*, an external optical system), surface tangibles get tracked by the system when they are in contact with the surface.

They enable precise manipulations [13, 15, 43], support eyes-free control [21, 45], can encourage specific collaboration strategies [2, 12] or increase awareness of others' actions [8]. Such tangibles have been considered to interact with horizontal surfaces, *e.g.*, [23, 33, 50], or with specific supporting structures [44]. But they cannot easily be used with large vertical surfaces as multiple tangibles become cumbersome when they cannot be left on the display.

In the non-digital world, people work with tangibles on vertical passive surfaces. Whiteboards with magnets, investigation boards with pins and threads, architect desks with drawing tools are all examples of tangibles on vertical or inclined surfaces. However, when it comes to vertical surfaces that actively emit light such as LCD or LED screens, the use of tangibles raises issues. Simple tools just fall, while pins and magnets damage electronics in screens.

In this article, we introduce *WallTokens* as a means to prototype tangible interaction for vertical surfaces (Figure 1). A *WallToken* is a light, passive tangible that is mounted on feet in order to generate a multi-touch pattern (*i.e.*, its footprint) when put in contact with the wall. A *push-handle*, located at the token's center, controls an additional foot which is equipped with a spring and terminated by a suction cup. Pushing this handle brings the suction cup in contact with the surface, attaching the token to the wall. Once attached, the token can be manipulated as a rotary knob, or left as is to free the user's hands for any other action. Users can easily detach a token with a simple lever movement that uses any of the peripheral feet as a pivot. This pulls the suction cup away from the wall, which makes the central foot contract back. Users can thus attach and detach tokens at will, making it easy to switch between different tokens or between barehand gestures and token manipulations.

In this article, we detail the design and fabrication process of *WallTokens*, which relies on 3D printing and basic supplies (*i.e.*, springs and suction cups) only. We describe how well they get recognized on our wall display, which is equipped with an infrared frame for detecting multi-touch input. In a study, we compare the performance of tangibles and multi-touch gestures for manipulating virtual objects, showing that participants' experience is better with tangibles than with multi-touch gestures. We finally report on our prototyping experience with a sample of applications that we developed on top of our library for handling token input.

Our main contributions are:

- a solution to enable tangible interaction for vertical displays;
- a low-cost approach to the prototyping of such tangibles;
- a study in which tokens performed better than multitouch gestures for manipulating virtual objects.

2 RELATED WORK

Our contribution is to enable tangible input in the context of vertical displays. We review previous work that discusses the individual benefits of vertical surfaces and tangibles for interaction, before detailing the very few projects that combine tangible input with vertical displays.

2.1 Horizontal vs Vertical Surfaces

A few studies have observed the effect of display orientation on users' experience. First, Morris *et al.* [34] have studied the impact of orientation for additional displays in an office environment for single users, and have reported that horizontal displays can cause ergonomic discomfort for some tasks such as reading long documents in comparison with vertical ones. A vertical orientation is thus preferable in some cases, particularly when the information space has a reference orientation.

In a collaborative context, Rogers and Lindley [41] conducted a study where teams of three participants had to solve a trip planning task. They compared three conditions: a desktop+mouse control condition, and two larger ($\sim 1\times 1$ meter) shared display conditions with a single pen controller that users have to exchange. The two shared display conditions differ in the display orientation: horizontal (table) or vertical (wall). Overall, the horizontal condition seemed to better promote collaboration between participants. A possible reason why users may have prefered the horizontal condition is the ease of both exchanging the pen and switching between paper-based and digital activities. In the vertical condition, participants complained about the fact that there was *no obvious place to put the pen down*. Interestingly, in Potvin *et al.*'s study [38] where each participant had their own pen and could keep holding it, there was no clear difference between the horizontal and vertical orientations for completing a design task collaboratively. These results suggest that tangible artefacts can be cumbersome when working with a vertical display if users cannot easily put them down somewhere.

2.2 Benefits of Tangibles

The HCI literature has proposed applications that rely on tangibles for collaborative interaction in many domains such as art [23], education [2], design [16] or document editing [51]. There are also empirical results that support tangibles' advantages in such contexts. For example, in Schneider *et al.*'s study [42], pairs of apprentices in logistics had to design a warehouse on a tabletop using either multi-touch interaction or tangibles. In such a high-level, problem-solving task, tangibles were preferred over touch gestures because they offered a better support to the exploration of alternatives. Antle *et al.* also showed that tangibles can be beneficial to the implementation of specific design strategies that enforce or encourage exchanges between users [2, 12]. Finally, Cherek *et al.* report on a study where participants were playing a game on a tabletop and had to react to others' actions while playing [8]. When performed with tangibles, actions were better detected than with virtual objects, suggesting that tangibles increase awareness of others' actions, which is key to collaboration [10, 14].

Tangibles have also been studied at a lower level, demonstrating their efficiency for fine-grained manipulations of virtual objects. In their seminal study, Fitzmaurice and Buxton [13] show that enabling *space-multiplexed* input with multiple specialized tangibles offers more precision than a *time-multiplexed* input device such as the mouse. Later, Tuddenham *et al.* [43] compared the performance of multi-touch and tangible input when interacting with virtual objects displayed on a tabletop. They considered both a simple docking task and a pursuit task that involves four objects as in Fitzmaurice and Buxton's study. In these studies, participants were faster and more accurate with tangibles than they were with multi-touch manipulations.

The advantages of tangibles have been further supported by other studies. Voelker *et al.* [45] report on a study where they compared physical knobs and virtual knobs (manipulated with one or two fingers). They found that physical knobs outperform virtual ones (in terms of both speed and accuracy), and that one-finger virtual knob performance degrades relatively more when used eyes-free than the other two conditions (whose performance remained mostly unchanged). Hancock *et al.* [15] propose tangibles with a mounted trackball that provides an additional control dimension. They report on an experimental task that consists of manipulations with multiple degrees of freedom (DOF). Compared to touch gestures, tangibles did not significantly differ for 5-DOF tasks, but were much faster for 3-DOF tasks. Participants also felt more precise when controlling a parameter with tangibles, in particular for the data exploration task that they studied. However, Hancock *et al.* also mention the occlusion issues that can occur with tangibles. Finally, Tangible Tiles [48] is a system that provides a collection of transparent tokens for manipulating digital images on a tabletop. In a task where pairs of participants had to explore images to find hidden features, manipulations with tangible tiles were still less efficient than manipulations of paper images, but they were more efficient than single-touch manipulations.

2.3 Wall Displays and Tangibles

As mentioned in the introduction, locomotion is key to interaction with wall displays. Users interact with the wall both from close-up and from afar [1]. This means that the design of tangibles for wall displays should consider (at least) the following two themes in Hornecker and Buur's framework of Tangible Interaction [18]: *tangible manipulation* to act on digital information through manipulation of material objects, and *spatial interaction* to take into account users' physical navigation in front of the wall.

Distant actions for interacting with a wall display can be performed with mid-air gestures (*e.g.*, [36]) and with physical objects such as personal devices (*e.g.*, [6, 47]) or custom-made objects that are manipulated in the air [3]. However, when it comes to interaction within arms' reach, *surface* tangibles have almost never been considered. Surface tangibles refer to physical controllers that users put or slide on the surface, as opposed to tangibles that are manipulated in the air (as the brain prop in [3]) or on another surface (as the sliders attached to a tablet in [21]). To the extent of our knowledge, the only example of such surface tangibles for a wall display is the Miners project [46] that makes use of tangible+touch interactions for a multi-player game. Such interactions proved very engaging for users. However, the use of tangibles remains underexplored, as each player manipulates a single token that is associated with a specific action. Furthermore, each player must always hold the token in their hand. Our contribution with *WallTokens* is to enable an interaction style where users can not only manipulate tokens on a vertical surface but also *leave them in-place* on that surface. This makes it possible for users to pin tokens at locations of interest, to switch between different tangibles or to simply free their hands when needed.

Many possibilities can be considered for fabricating tangibles. In the context of horizontal displays, solutions include optical sensing to track fiducial markers attached to the tangibles (*e.g.*, [23, 48]), capacitive sensing with tangibles that create specific multi-touch patterns when in contact with the surface (*e.g.*, [33, 50]), magnetometers or Hall sensors with magnetic tokens [4, 27, 28], or even more elaborate electronics with, *e.g.*, bluetooth communication [9]. But, whatever the technology considered, there is no obvious way of making it work on as is with a vertical surface. The main issue to address is that of gravity, which prevents users from dropping tangibles without them falling to the ground.

3 WALLTOKENS

WallTokens are tangibles that are low-cost and easy-to-fabricate, making them ideal for prototyping. They consist of basic supplies (spring, suction cup and felt) and 3D printed parts assembled together. They are passive, designed to be interacted with on multi-touch surfaces. Like some previous projects (*e.g.*, [5, 21, 25, 33]), each *WallToken* is mounted on feet that generate a multi-touch pattern when in contact with a tactile surface. A pattern is specific to a token, making this token recognizable with a pattern matching algorithm.

3.1 Fabrication

A *WallToken* consists of several modular components that are then assembled together by means of screwing and interlocking. Figure 2-left details these different components.

- The **base** ④ is the main component. It consists of a plate with three feet underneath (*i.e.*, a single 3D printing job). When in contact with the wall, the three feet will generate the multi-touch pattern (*i.e.*, the token footprint) that is specific to the token. Each foot is 25 mm tall and 12 mm wide. In order to avoid any scratch on the surface, each foot is carved with a placeholder where a piece of felt can be glued (each placeholder is a 4 mm side square of 1 mm depth). The plate is 3 mm thick. Its shape can vary (*e.g.*, square, circle, triangle as illustrated in Figure 2-right). The plate not only gives the token a visually identifiable shape, it also prevents users' fingers from getting too close to the screen and thus interfering with the token footprint during the recognition process.

- The **grip** ③ is the knob where users put their fingertips to hold and manipulate the token. It is connected with the base thanks to a snap fit system. The grip is a 20 mm high cylinder of 4.85 mm radius at its base. Its contour is slightly curved to make it comfortable.

- The **push-handle** ① is the top of the token. It is under the palm of users' hand when they hold the token. It is 15 mm long from the base to the tip, and the radius of its circular base is 16 mm. Users push this handle with their palm when they want to attach the token to the surface. To detach a token, users pull its grip. This creates a lever effect around one of the token's feet, making this action easy to perform.

- The **central rod** ⑤ is a 61.5 mm tall stick that connects the **suction cup**, which can attach the token to the surface, to the push-handle. By default, the suction cup is not in contact with the surface. The bottom of the central rod is designed as a placeholder where the suction cup can be snapped. The top of the central rod ends with a thread on which the push-handle can be screwed.

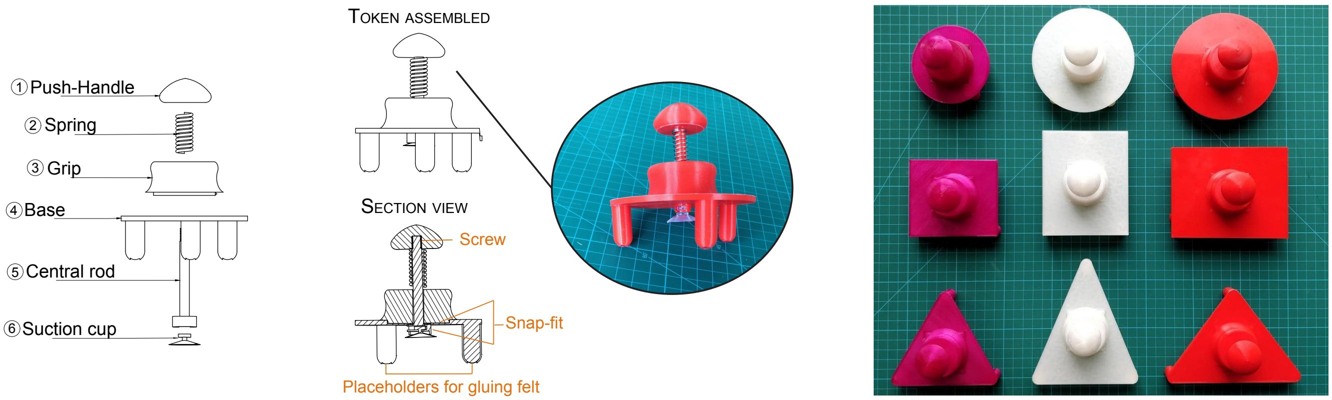

Figure 2: (Left) A schematic representation of a *WallToken*'s components, and a real token. (Right) The 9-token final set tested in the Recognition Experiment. Real tokens are laid on a surface textured with 1 cm squares.

- The **suction cup** ⑥ is a standard 20 mm diameter one, which generates a suction force of approximately 130 newtons. When the central rod is at its default height, the suction cup does not touch the screen, remaining 15 mm away from it.
- The **spring** ② maintains the central rod at its default height when the suction cup is not attached to the wall. It has 10 coils and is made of steel (AISI 304L stainless steel). Its length is 25 mm when free, 7 mm when compressed. Its inner diameter is 0.7 mm, and its outer diameter is 0.9 mm.

WallTokens are low-cost, yet robust. They do not require any electronics, but only passive materials. Apart from the suction cup, the spring and the felt, individual components are fabricated with a 3D printer using PLA or Tough PLA filament.[1] The assembly time for a token is less than a minute once the felt has been glued under the feet. We ran some informal tests to assess how long tokens can stay on the wall with such a design. Our tests revealed that they do not fall off for at least twelve hours when the wall is off, and for at least three hours when it is on (heat has an impact on how well the suction cups stick). *WallTokens* also proven quite robust against repetitive manipulations. In particular, we used the same unique token for the experiment that we report in section 4. However, in case a component gets broken, the modularity of the fabrication process makes it easy to replace the damaged part only.

Finally, modularity also makes it easy to test different token appearances, thus compensating for the lack of flexibility tangibles usually suffer from when it comes to customizing their appearance (*e.g.*, their shape or color) [42]. For example, interaction designers can test different *bases* during the design phase of an application. Having designed tokens ourselves, we also found modularity very convenient for deciding on ergonomic details. In particular, we have tested several alternatives for the *push-handle* and *grip* components before choosing the ones that were the most comfortable during manipulations. SDF files for fabricating the different components are available as supplemental material. Others can easily use them as is, or edit them to test alternative shapes for the base, grip and push-handle.

[1] We used Ultimaker 3/3X 3D printers.

3.2 Recognition

As mentioned above, we rely only on multi-touch tracking in order to recognize a token based on its footprint using the pattern matching algorithm from [33]. A token set is represented by a collection of templates, with one template per token. Then, when at least three contact points occur simultaneously,[2] they are processed with the pattern matching algorithm in order to identify the closest template. Relying on such a recognition strategy means that all tokens' footprints should be different from one another within a token set.

When a token is in contact with the wall, any contact point occurring inside the triangle formed by the token feet triggers the detection of the attached state of that token. The token's state remains set to `Attached` until a token-up event occurs. Such an event occurs as soon as two token feet are lifted off the surface.

Although *WallTokens* could be made conductive to work with a capacitive surface, very large tactile surfaces usually rather rely on optical tracking for detecting touch input. We designed *WallTokens* to interact with such a very large wall display, which is made of tiled ultra-thin bezel screens. It is equipped with a PQlabs infrared touch frame, which is located 4.5 mm in front of the screens. Detailed specifications for such a technology are not available. We thus had to conduct a series of empirical tests to assess 1) the minimal diameter of a foot to be detected as a touch point (12 mm), 2) the minimal height that feet should have to keep the token base and the suction cup when in its default state out of tracking range (25 mm), and 3) the minimal distance between two feet to avoid getting them merged as a single touch point (D_{min}=5 mm).

The tests mentioned above also revealed occlusion issues related to the infrared technology, which impacted our strategy for designing tokens' footprints. A foot, which is both aligned with a second foot along the x-axis and with a third foot along the y-axis, is not detected. In order to minimize the chances for such an issue to occur, the three feet of a token always form an isosceles triangle. Figure 3 illustrates feet configurations for the nine tokens in our set. The smallest footprint is a 58 mm side equilateral triangle. It corresponds to the smallest footprint that ensures a minimal distance

[2] Contact points should occur within 200ms, and should be close enough to the recognizer's best match.

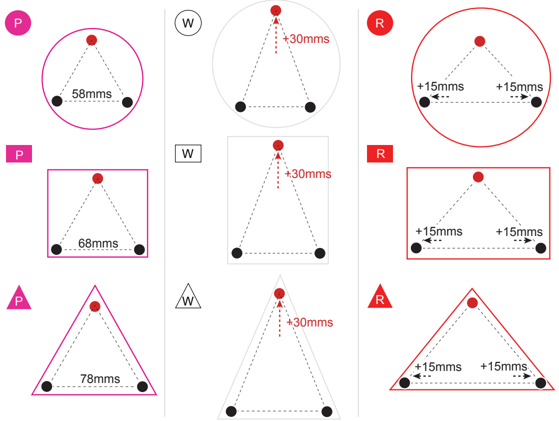

Figure 3: Footprints for the nine tokens in our final set. Pink tokens' footprints form equilateral triangles. White tokens are derived from pink tokens by pulling one vertex from 30 mm. Red tokens are derived from pink tokens too by pulling two vertices from 15 mm each in order to make the basis 30 mm larger.

of D_{min} between two feet when they get projected on the x- or y-axis regardless of the token orientation.

The position of the central rod is set in order to optimize the detection of token states (`Attached` or `Detached`). State detection accuracy results from a trade-off between maximizing the distances from the central rod to the tokens' feet and avoiding the potential occlusion issues mentioned above. Depending on the token's feet relative placement, a good compromise for positioning the central rod is either the center of the circumscribed circle of the triangle formed by the token's feet (pink and white tokens in Figure 3), or the center of the inscribed circle (red tokens in Figure 3).

We ran a small-scale experiment in order to validate *WallTokens*' design. We both consider recognition accuracy of tokens' identity and detection accuracy of their state (`Attached` or `Detached`). In order to get ecological observations, our experimental task collects measures in the context of *pick-and-drop* interactions (*i.e.*, users put a tangible controller in contact at two locations consecutively) and of *detach-then-attach* interactions (*i.e.*, users move a tangible mark from one location to another).

3.2.1 Participants. Because of the COVID-19 pandemics , only the three authors of this submission participated in this experiment: one woman (40 year-old) and two men (27 and 53 year-old).

3.2.2 Apparatus. The experiment runs in full screen mode on a wall-sized display (75 ultra-thin bezel screens tiled in a 15×5 grid, resulting in a total surface of 5m90 $\times$ 1m95 for a resolution of $14\,400 \times 4\,800$ pixels), driven by 10 workstations[3] and equipped with a multi-touch PQLabs frame connected to a workstation. The experimental software was developed using Unity 3D (version 2018.3). The whole setup is orchestrated by a laptop workstation.[4]

[3]Dell workstations with a 3.7 GHz Quad Core Intel Xeon CPU, a NVIDIA Quadro K5000 GPU and 32 GB RAM running Linux.
[4]MSI GE72 2QF laptop with a 2.90 GHz Intel Core I5 CPU, a NVIDIA GeForce GTX970M GPU and 16 GB RAM running Windows 7.

3.2.3 Task. As Figure 4 illustrates, our experimental task consists of several steps in order to challenge the recognizer in different contexts. 1 A colored shape stimulus is displayed on the wall at 1.25 m above the floor in front of the user. Participants have to grab the token whose shape and color match that of the stimulus (*e.g.*, the pink triangle token in Figure 4) and put it in contact with the wall at the stimulus' location. This makes the stimulus disappear, and another stimulus appear 60 cm to the right. 2 Participants have to put the token in contact with the wall at this new location. 3 The stimulus' texture turns into a checkered pattern after a 1-to-2 s random delay. As soon as the texture changes, participants have to attach the token to the wall as fast as possible. 4 - 5 In order to make sure that the token is actually attached to the wall, participants have to release the token to touch a green circle that appears close to the token. They then have to put their hand back on the token to be ready to detach it. 6 The stimulus' fill texture turns into a checkered one after a 2.5 to 3.5 s random delay, instructing the participant to detach the token from the wall as fast as possible. 7 A final stimulus appears 30 cm to the left. It is filled with a checkered pattern as soon as it appears, indicating to participants that they should place the token in this location and attach it as fast as possible.

3.2.4 Design and procedure.

Factor and Design. The only factor is TOKENTARGET. It can be one of the nine tokens of our final set (Figure 2-right): *RedCircle* R , *RedRectangle* R , *RedTriangle* R , *WhiteCircle* W ,*WhiteRectangle* W , *WhiteTriangle* W , *PinkCircle* P , *PinkRectangle* P , *PinkTriangle* P . During the experiment, each participant has to complete one block per TOKENTARGET, each block consisting of four replications of the experimental task described above. We use three different presentation orders for the nine blocks, one per participant. This design results in 108 completed tasks in total (3 participants $\times$ 9 TOKENTARGET $\times$ 4 replications).

Measures. The token recognized by our algorithm is logged at steps 1 , 2 and 7 , allowing us to compute accuracy of token recognition (*RecognitionAccuracy*) over $108 \times 3 = 324$ observations. Regarding state detection, the green circle appears after step 3 only if our algorithm actually detects the `Attached` state. In case it does not, the current task is canceled (with already recorded measures ignored), the operator counts an error (*StateDetectionError*) and restarts the task. *StateDetectionError* is incremented each time such an error occurs. The experiment software additionally logs two time measures: *AttachTime*, which is the time between the first texture change and the `Attached` state detection that follows (step 3), and *DetachTime*, which is the time between the second texture change and the token up event that follows (step 6).

Procedure. Participants stand about 45 cm in front of the wall. The nine tokens are available on a table to their left. The session of tasks begins with a series of nine practice tasks, one task per token. Participants are allowed to redo the practice set until they feel comfortable enough. They then perform the 36 measured tasks.

3.2.5 Results. Table 1 reports the overall recognition accuracy, as well as a break-down per participant (A1, A2, and A3) (means and 95%-CIs).

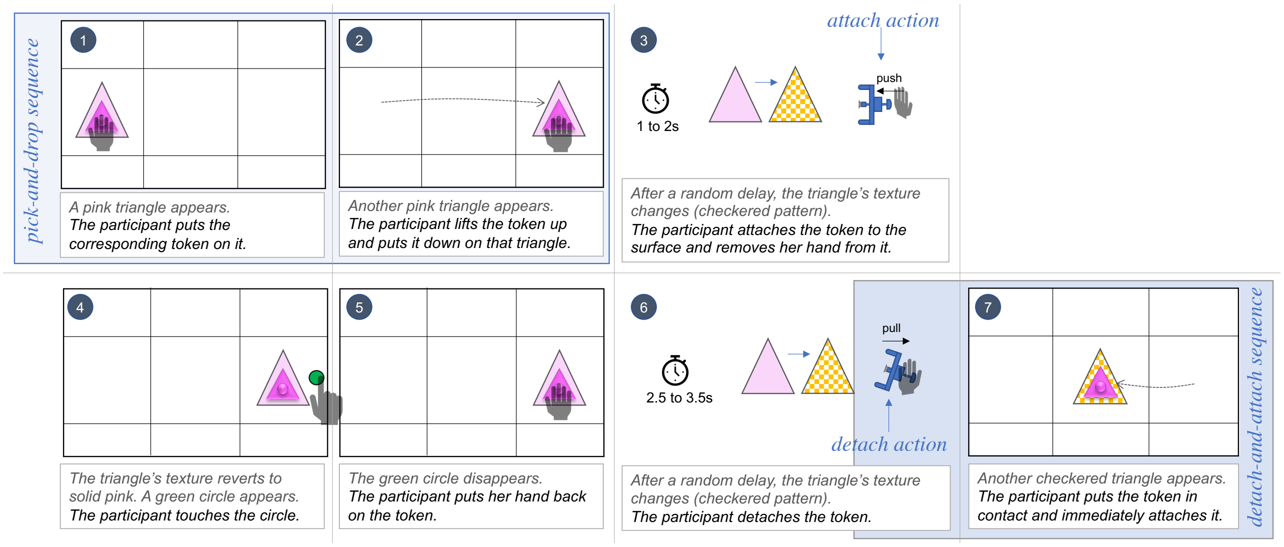

Figure 4: Recognition Experiment: main steps of a trial.

Participant	*RecognitionAccuracy* (%) 1 + 2 + 7	*StateAccuracy* (%) 3 + 7	*AttachTime* (ms) 3	*AttachTime* (ms) 7	*DetachTime* (ms) 6
A1	100 ± 0.0	100 ± 0.0	628 ± 14	852 ± 46	797 ± 41
A2	95.4 ± 4.6	98.6 ± 2.7	742 ± 55	1197 ± 106	944 ± 41
A3	98.1 ± 2.5	100 ± 0.0	674 ± 47	880 ± 43	770 ± 26
Mean	97.8 ± 2.6	99.5 ± 0.9	681 ± 65	976 ± 218	837 ± 107

Table 1: Results of the recognition experiment.

Overall, seven token recognition errors occur among 324 measures. Two occurred at step 1 (both for participant A2), five at step 7, and none at step 2. There was no systematic pattern regarding the confusion errors. This prevents us from drawing conclusions regarding the design of the tokens, as errors might as well result from the touchframe itself, which delivers noisy input on some rare occasions. Regarding state detection, there was only one error among 216 measures. This error occurred with the smallest token (PinkCircle), which happened for A2 at step 3.

Time measures suggest that attaching and detaching a token can be performed in less than a second and that it can take an expert user as little as 500 ms to attach a token already in contact with the surface (time at 3 includes reaction time, and time at 7 additionally includes a movement), and 650 ms to detach a token (6 includes reaction time). Comparing time for attaching at step 3 with time for detaching suggests that it is easier to attach a token than to detach it. This matches our initial impressions.

We acknowledge the limited ecological validity of these observations as they come from the authors themselves. These results cannot be generalized to average users. However, they give an indication of the performance envelope [11], *i.e.*, the performance that expert users can reach.

4 INTERACTING WITH WALL DISPLAYS: WALLTOKENS VS TOUCH GESTURES

WallTokens can act as controllers on a wall display for, *e.g.*, manipulating virtual objects or adjusting parameter values. We believe that they can be an efficient alternative to touch gestures when interacting with a wall from close distance. In this second experiment, we evaluate the performance of *WallTokens* when they are used as controllers on a wall display.

We hypothesize that *WallToken* manipulations are more efficient controllers on a wall display than touch gestures are. First, previous studies have shown that users are more accurate with tangibles than with bare finger input [15, 45]. We thus hypothesize that users will be faster with *WallTokens* than with touch gestures (H_1). Second, we hypothesize that a *WallToken* is more comfortable than touch gestures are for continuous manipulations (H_2). This is not only because of the felt under their feet that reduces friction with the wall, but also because of the possibility for users to reposition their fingers on the token while manipulating it. Finally, we also hypothesize that *WallTokens*' relative advantage will be lower when manipulating small virtual objects which are under their base than when manipulating large virtual objects (H_3). This is because tangibles cause more occlusion than bare hand gestures.

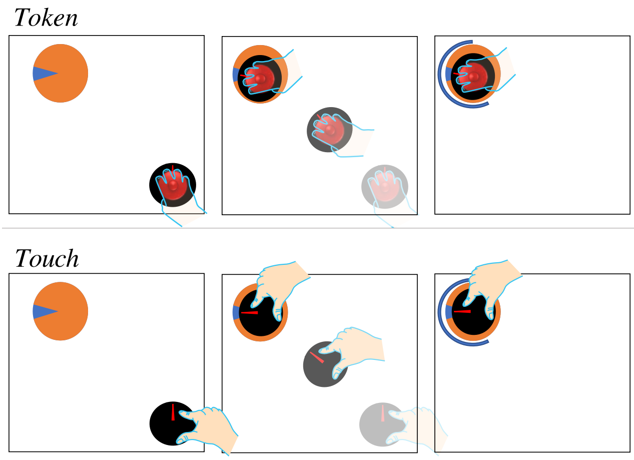

Figure 5: The task under the two INPUT conditions in experiment *Walltokens vs Touch Gestures*.

4.1 Participants

Twelve volunteers (9 men and 3 women), aged 24 to 44 year-old (average 28, median 25.5), participated in the experiment.

4.2 Apparatus

The apparatus is the same as the one described in Section 3.2.2.

4.3 Task

Participants had to perform a docking task, where they had to manipulate a virtual object (Modulus) to make its position and orientation match that of a target placeholder (Stimulus). They had to perform such a task using either *WallToken*-based manipulations or common multi-touch gestures (Figure 5).

The task starts with the two objects displayed on screen: the Modulus as a black circle, and the Stimulus as an orange circle. Depending on the condition, the participant has to put his fingers or the token on the Modulus and drag it over the Stimulus. The task ends when the Modulus has been maintained for 1500ms inside the Stimulus with its orientation (indicated by a red line) matching that of the Stimulus. As soon as the position and orientation conditions are met, a blue ring starts to fill up. The ring is full when both conditions have been maintained for 1500ms (dwell), ending the task. The experiment software allows for some tolerance in both orientation and position. The difference in orientation between the Stimulus and the Modulus should be less than 10° and the distance between their centers should be less than 1cm.

Contrary to Tuddenham *et al.*'s study [43], which also compares multi-touch with tangibles, our experimental task involves a single tangible. This is because we do not advocate for the use of *WallTokens* for spatial multiplexed input where one user would manipulate several objects concurrently with frequent switches between multiple *WallTokens*. The cost of repetitive attach and detach actions would be too much of an overhead. In a wall display context, we rather envision the use of tangibles as controllers for longer interactions. For example, when each user has their own tangible, or when interactions with a given token are performed in sequence. Our experimental task operationalizes such interactions.

4.4 Design and procedure

Factors. Our experiment involves the following three primary factors:

- INPUT ∈ {*Token*, *Touch*}. The *WallToken* used in the *Token* condition was a red 9.4 cm-diameter circle. We chose a circle shape so that the amount of occlusion does not depend on the token's orientation. In the *Touch* condition, users perform 2-finger gestures to control both position (*i.e.*, middle of the 2-finger segment) and orientation (*i.e.*, orientation of the 2-finger segment).
- SIZE ∈ {*Small*, *Medium* or *Large*}. This controls the relative size of the Modulus relative to that of the token, and thus the amount of occlusion caused by the token (*i.e.*, the smaller the object, the greater the occlusion caused by the token). *Small* (resp. *Large*) means that the Modulus' diameter is half (resp. twice) that of the token, and *Medium* means that the Modulus and the token have the same size.
- ROTATION ∈ {0°, -90°, 90° and 180°} corresponds to the Stimulus' orientation.

The fourth factor, DIRECTION, is a secondary one that we introduced for ecological purposes. It specifies how the Stimulus is displayed relative to the Modulus when the task starts. It can take the following four cardinal directions: *NW*, *NE*, *SW* or *SE*. When the task starts, the Modulus is displayed within users' arm reach (within a 39 cm square at 1.05 m height), and oriented along the y-axis. Its precise location depends on the value of DIRECTION in order to ensure that both objects are displayed in the same screen. This is to avoid users having to cross bezels between the wall's screen cells, as they are an artifact of the specific wall prototype we use in this experiment. The distance between the Modulus and Stimulus is always 30 cm, but their relative positions depend on DIRECTION.

Design. Trials are blocked by INPUT. Each block contains three sub-blocks, one per SIZE. Each sub-block consists of 32 trials, *i.e.*, each DIRECTION × ROTATION combination, and is replicated twice. Presentation order of block is counterbalanced across participants, and presentation order of series of sub-blocks is counterbalanced across participants and INPUT conditions. Within a sub-block, trials are presented in a random order. This design results in 2304 trials in total: 12 participants × 2 INPUT × 3 SIZE × 4 DIRECTION × 4 ROTATION × 2 replications.

Measures. We collect the following measures: 1) *Time*, the task completion time (*i.e.*, the timer starts as soon as the token or two fingers touch the wall and stops when the 1.5 s dwell ends); 2) *clutchActions*, the number of times the token or participants' fingers leave the wall during the task; and 3) the Modulus' position and orientation at each input event.

Procedure. Participants first sign a consent form. They then stand in front of the wall at a distance of about 45 cm from it. The operator gives instructions for completing a task.

Each INPUT block is preceded by three practice sub-blocks, one per SIZE, with each sub-block containing four trials (*i.e.*, four DIRECTION × ROTATION conditions randomly taken out of the sixteen combinations). Participants can request to do the practice session again if they do not feel comfortable enough. They then complete

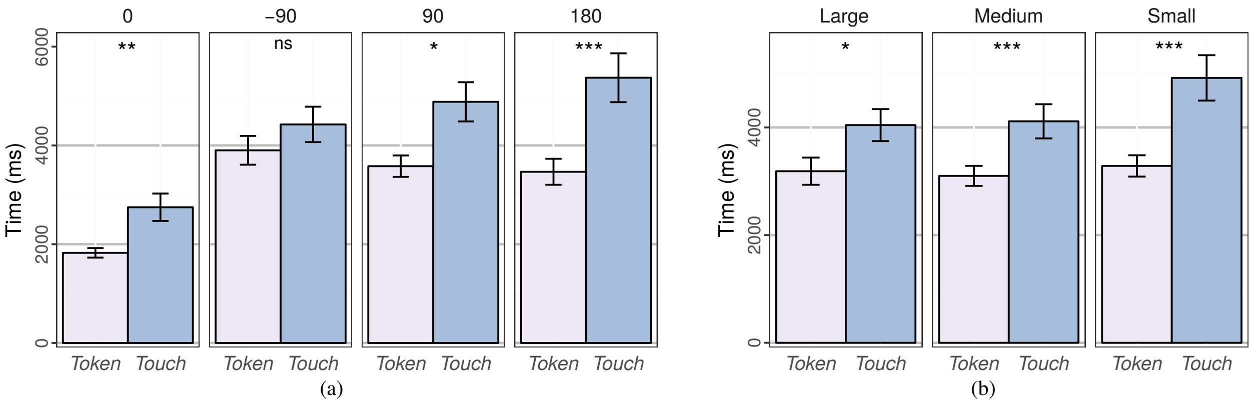

Figure 6: (a) Completion time by INPUT × ROTATION; (b) Completion time by INPUT × SIZE. Error bars represent the 95% confidence interval. *Token* is faster than *Touch* for each ROTATION and SIZE condition. However, the difference is not significant for the −90° ROTATION condition as shown in the bar charts ('*' < 0.001, '**' < 0.01, '*' < 0.05, 'ns' otherwise).**

the 97 logged tasks of the first INPUT block. Before proceeding to the second block, they have to sit and rest their arm until they do not feel any more fatigue. In all cases, the break cannot be shorter than one minute.

At the end of the experiment, participants fill in a questionnaire where they grade on a 5-point Likert scale each INPUT along the following aspects: physical demand, mental demand, comfort, performance and occlusion. They then have to circle their preferred INPUT technique for each SIZE condition and overall. The operator also collects participants' informal feedback during this debriefing phase. The whole procedure lasts about 45 minutes.

4.5 Results

Completion Time. We first analyze the effect of our primary factors[5] on completion time (*Time*) using a repeated measures ANOVA for the model INPUT × ROTATION × SIZE, and Bonferroni-Holm corrected paired post-hoc t-tests. Figure 6 illustrates our results.

First, INPUT has a significant effect ($F_{1,11}$=35.5, p<0.001, η^2_G=0.17), with all participants being consistently faster with *Token*, and *Token* being about 26% faster than *Touch* on average. This result supports H_1 (*i.e.*, users are faster with *WallTokens* than with touch gestures).

We also observe a significant effect of ROTATION on *Time* ($F_{3,33}$=56.2, p<0.001, η^2_G=0.33). Unsurprisingly, participants were significantly faster in the translation-only condition (0°) than in all other conditions (−90°, 90° and 180°, all p's < 0.001). However, there is no significant difference between conditions −90°, 90° and 180° (p's > 0.4).

Interestingly, there is a significant INPUT × ROTATION interaction on *Time* ($F_{3,33}$=6.78, p=0.001, η^2_G=0.04). As Figure 6-(a) illustrates, the −90°, 90° and 180° conditions are not significantly different in the *Token* condition, while participants have been significantly faster with −90° rotations than with 180° rotations in the *Touch* condition (p=0.016, and no significant difference between pairs <90°,180°> and <90°,−90°>). This is consistent with previous experiments (experiment 2 in [52], and lateral condition in [17]) where participants were faster for clockwise rotations (−90°) than for counterclockwise rotations (90°) with touch gestures. This might be because of a lower cost of movement planning [37] for right-handed users for clockwise rotations than for counterclockwise rotations. In comparison, *WallTokens* are less sensitive to differences in orientations. This results in *Token* being faster in all conditions, but not significantly in the −90° condition (p=0.072).

Finally, there is a significant effect of SIZE on *Time* ($F_{2,22}$=7.79, p=0.003, η^2_G=0.03), with participants being significantly slower with *Small* than they were with both *Medium* and *Large* (p's < 0.015). This suggests that occlusion caused performance issues for *Small* targets. However, contrary to what we hypothesized, the interaction INPUT × SIZE is not significant ($F_{2,22}$=2.29, p=0.124, η^2_G=0.01). Even more surprising, not only were participants significantly faster with *Token* than with *Touch* for all SIZE conditions (p < 0.001 for *Small* and *Medium*, p=0.014 for *Large*), but they seem to have been even more relatively faster when they had to manipulate small-sized objects (Cohen's d=1.39 for *Small*, d=0.99 for *Medium* and d=0.91 for *Large*). These results reject H_3: occlusion issues are not more detrimental with *WallTokens* than they are with touch gestures.

Clutching. Our second measure, *clutchActions*, gets incremented each time either the fingers or the token loose contact with the wall during a task (*i.e.*, they clutch to adopt a more comfortable posture in order to keep on controlling). As collected data do not follow a normal distribution, we use paired Wilcoxon signed rank tests for statistical analyses.

The average number of clutch actions is significantly lower in the *Token* condition than in the *Touch* condition (0.11 ± 0.04 vs 0.88 ± 0.16, p<0.001), and this difference is consistent across different ROTATION conditions (p's < 0.003) and SIZE conditions (p's < 0.002).

Interestingly, *clutchActions* hardly varies between the different SIZE levels in the *Token* condition (p's > 0.41), while it significantly grows when SIZE decreases in the *Touch* condition (from 0.68 ± 0.25 to 1.11 ± 0.33 with significant difference between *Small* and *Medium* and *Large*, p's < 0.027). This increasing need for repositioning their fingers with small objects might explain the relative disadvantage of touch gestures compared to tokens that gets higher in the SIZE=*Small* condition. Similarly, there is no significant difference in *clutchActions* between the different ROTATION levels for

[5]The ecological factor DIRECTION has no effect on *Time* and no interaction effect with any of the other three primary factors.

Token, while some differences are significant for *Touch*. *clutchActions* is significantly lower for 0° (0.29 ± 0.18) than for the other ROTATION levels (p's < 0.004, 0.84 ± 0.30 for $-90°$, 1.05 ± 0.39 for 90°, and 1.32 ± 0.36 for 180°), and $-90°$ has also significantly less *clutchActions* than 180° (p=0.009). Here again, this seems to be related to time performance as 0° and $-90°$ are the conditions where participants performed best for *Touch*. Overall, the need for finger repositioning in the *Touch* condition seems to have a negative impact on users' performance. In comparison, *WallTokens* allow users to reposition their fingers individually on the token while keeping it in contact with the wall, allowing greater fluidity in control.

Integrality & Simultaneity. A movement that affects several dimensions (here translation and rotation) is integral if the movement can concurrently modify the value of the different dimensions [19]. Integrality gives an indication of the fluidity in the control of several dimensions, and is thus interesting for comparing touch gestures and token manipulations at a fine-grained level. There is no standard way to measure integrality. Here, we adapt the method from [19] with recommendations from [32] by considering as integral a portion of movement where both dimensions change simultaneously to get closer to their target values. Our *Integral* measure is thus the percentage of movement time during which the differences in position and orientation between the stimulus and the modulus decrease each by more than a given threshold th, or these differences are both very small (*i.e.*, the movement is *stable* as is typically the case during clutching actions and small adjustments at the end of the movement). The steps for computing *Integral* are as follows:

- we consider the movement from the trial start time to the first time the docking conditions are met (dwell start time), and segment it into 10 ms intervals. For each interval, we compute the difference in position and orientation, Δ_{pos} and Δ_{orient} (normalized in $[0,1]$);
- we smooth data to remove sensor noise (using R's `smooth.spline` function with default parameters);
- we classify each interval as: (i) *integral*: $\Delta_{pos} \geq th$ and $\Delta_{orient} \geq th$; (ii) *stable*: $-th < \Delta_{pos} < th$ and $-th < \Delta_{orient} < th$; (iii) *separate*: neither integral nor stable;
- then, we compute the percentage of stable and integral intervals among all intervals.

With th=0.001 (0.1% of the movement amplitude),the average values for *Integral* are $41.9\% \pm 1.3$ for *Token* and $29.8\% \pm 1.2$ for *Touch*. The difference is significant ($p < 0.001$), as differences per positive ROTATION condition are. This suggests that *WallTokens* enable manipulations that are more fluid than touch gestures do, allowing users to manipulate position and orientation in an integral movement.

Qualitative Feedback. Figure 7 reports participants' qualitative feedback by INPUT. In the final questionnaire, they had to give grades for: comfort, mental demand, occlusion issues, perceived performance and physical demand. We compare these 5-point grades with paired Wilcoxon signed-rank tests.

Token receives better scores regarding comfort (p=0.002) and physical demand (p=0.008). This supports H_2 (*i.e.*, *WallTokens* are more comfortable than touch gestures are for continuous manipulations). *Touch* actually causes more friction and clutching actions than *Token* does. Furthermore, touch detection through optical technology (such as the infrared frame on our wall) can cause more discomfort

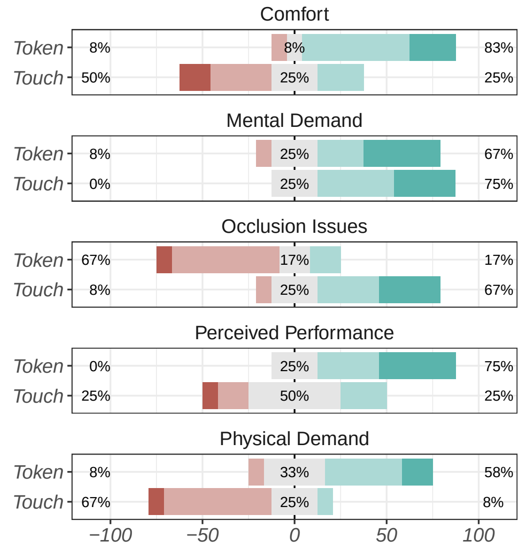

Figure 7: Distribution of grades (on 5-point Likert scales) for each INPUT along the dimensions: Comfort, Mental Demand, Occlusion Issues, Perceived Performance and Physical Demand (positive assessments are green and on the right).

than *e.g.*, capacitive screens as users may have to adopt hand postures that prevent other fingers from entering the tracking range of the frame. Perceived performance is consistent with actual performance, with *WallToken* outperforming *Touch* (p=0.014). Conversely, grades regarding occlusion issues are in line with our hypothesis H_3, with participants reporting tokens to be causing more occlusion than touch gestures (p=0.001). However, they are not consistent with quantitative observations as *WallTokens* did not perform worse than touch gestures with small objects. Finally, participants did not grade any of the two INPUT techniques as mentally demanding (p=1).

Summary. Overall, participants were faster with a *WallToken* controller than with bare hand gestures for manipulating a virtual object displayed on the wall. Participants found touch gestures less comfortable than token-based manipulations, the latter enabling more fluid and integral movements. However, tokens also have some drawbacks in comparison with touch gestures. First, although users' performance did not degrade more with tokens than with touch gestures when the amount of occlusion increases, participants still found occlusion more hindering with tokens than with bare hand manipulations. We are currently working on the design of *WallTokens* with a translucent base in order to address such issues. Second, the touch resolution of our research equipment might be lower than that of smaller, commercial touch devices. The large touch frame, which has been custom-built for our wall display, does not come with detailed specifications. However, our empirical tests reveal the following limitations: 1) distinct touch points should be distant from at least 5mm in order not to get merged into a single point, and 2) accidental touch events can get triggered because of other fingers that are too close to the wall (less than 4.5mm). It would be interesting to replicate our experiment with other hardware setups.

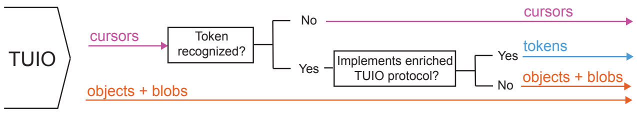

Figure 8: Enriching the TUIO protocol to dispatch token-related events.

5 APPLICATIONS

Tangibles can play a lot of roles when interacting with a digital surface. They can materialize users, data and actions to act as identifiers, containers, filters, queries, commands, etc. Many of these different roles are particularly relevant when interacting with a wall display. First, as our experiment above suggests, tangibles can act as controllers when interacting with a wall within arm's reach. They can act as specialized controllers, freeing users from having to learn complex touch gestures. They can also be used for moving content between distant locations using pick-and-drop interactions [40, 48]. Second, tangibles are particularly useful in collaborative contexts. For example, associating each user with a specific tangible is a low-cost and robust way of identifying individual users (as in, *e.g.*, [51]). This user identification method avoids relying on external optical systems, which are usually costly and vulnerable to occlusion issues that frequently occur in an environment where users physically move around. Moreover, tangibles avoid direct touch with the surface, and can thus limit potential sanitary issues when multiple users interact with it. Finally, wall displays are very large surfaces that raise challenges regarding workspace management. Tangibles can help in that regard as they can be used to mark specific positions. For example, they can be used for tagging personal areas as opposed to shared areas, or for bookmarking positions of interest.

5.1 Development Framework

To facilitate the development of applications involving *WallTokens*, we have developed a framework based on TUIO[6] [24], a widely used protocol for programming Tangible User Interfaces. Client libraries for TUIO are available in most programming languages (`C++`, `C#`, `Java`, *etc.*). They connect to an input server and dispatch events according to the TUIO protocol, *i.e.*, in the form of messages that consist of one action among {`down`, `update`, `up`} and the object that triggered this action. The protocol proposes three types of objects: `TuioCursor` (*e.g.*, a touch point along with its position); (ii) `TuioObject` (*e.g.*, a tangible along with its position and orientation); and (iii) `TuioBlob` (*e.g.*, an elliptic shape that often corresponds to the contact area of a `TuioCursor` or a `TuioObject`).

We developed a `C#` library that enriches the standard TUIO protocol with a new type of objects, `TuioToken`. The library connects to a standard TUIO server (*e.g.*, the PQLabs driver that runs our wall touch frame). As Figure 8 illustrates, it runs the *WallTokens* recognizer each time at least three `TuioCursor-down` events occur simultaneously. In case a token is recognized, it dispatches a `TuioToken-down` event. Otherwise, it passes on the three initial `TuioCursor-down` events. Then, any `TuioCursor-update` event from a cursor that has been recognized as being part of a token is turned into a `TuioToken-update` event until the three cursors leave the surface (which triggers a `TuioToken-up` event). Any `TuioToken` event consists of its action type ({`down`, `update`, `up`}), as well as the token's identity, contour shape, color and state. In addition, each time a cursor down (resp. up) event occurs within the envelope defined by the three token feet, the library dispatches a `TuioToken-update` event to communicate the token's `Attached` (resp. `Detached`) state.

Any `C#` object can implement the `TuioWallTokenListener` interface to listen to such `TuioToken` events with dedicated callback methods (`addTuioWallToken`, `updateTuioWallToken` and `removeTuioWallToken`). In order to also support any programming language, we developed a *WallToken* TUIO server that any application can connect to in order to receive enriched TUIO events. As illustrated in Figure 8, this server can either dispatch `TuioToken` events as described above, or it can downgrade these events into `TuioObject` and `TuioBlob` events, or even simple `TuioCursor` events. Such a downgrade makes it possible to run any existing TUIO-based application with *WallTokens* without modifying its source code. For instance, the *Map* application described in the next section was anterior to the *WallTokens* project. It was developed in `Java` with input based in part on touch `TuioCursors`. In order to enable *WallToken* input, we first ran the server in downgraded mode so that the *WallTokens* are considered as simple `TuioCursors`. This enabled *WallToken* input without writing any single line of code. We then progressively added callbacks specific to `TuioTokens` in order to handle events such as token rotations and attachments.

5.2 Demo Applications

We used the framework described above to develop three demo applications (Figure 9): a picture classification application (*Picture*); a map application (*Map*); and an artistic demo (*Art*). The *Picture* application displays a collection of pictures for users to classify by grouping them spatially. Large vertical displays are actually good at supporting the classification, grouping and comparison of visual components [20, 31, 39]. The *Map* application is a multiscale interface that allows users to pan & zoom a large map and instantiate multiple DragMags [49] to magnify specific areas. Finally, the *Art* application displays a water texture on the entire wall and lets users generate waves interactively, creating an aesthetically-pleasing rendering. This art demo was inspired by the ReactTable [23], which combines tangibles and a tabletop into an electronic instrument for collaborative musical performances.

Rather than providing a full description of each of these applications, we list below some of the roles that *WallTokens* play in them:

- ***WallTokens* as data containers**. In the *Picture* (resp. *Map*) application, a *WallToken* can be used to *pick-and-drop* [40] a picture (resp. a DragMag) from one location to the other. *Pick-and-drop* interactions are particularly important with large displays as dragging an object over a long distance quickly causes discomfort.
- ***WallTokens* as controllers**. In the *Map* application, sliding a *WallToken* pans the map while rotating it adjusts the zoom factor. Depending on the *WallToken*'s location, such pan & zoom operations apply to the whole map or are restricted to a specific

[6] https://www.tuio.org/

Figure 9: Application demos: picture classification, map navigation, artistic performance.

DragMag. In the *Art* demo, users can attach a *WallToken* to the wall to make water drops fall on the large water surface, and then rotate it to adjust the frequency at which drops fall.

- ***WallTokens* as cursors**. In the *Picture* application, users can perform *drag-and-drop* operations to adjust a picture's position. In the *Art* demo, users can drag a *WallToken* to cut through the water surface and generate waves.
- ***WallTokens* as bookmarks or pins.** Attaching a *WallToken* to the wall in the *Picture* application lets users change how they populate selections. Once attached to the wall, a finger swipe gesture that is initiated on a picture and oriented toward a *WallToken* actually adds the picture to the selection that is associated with that token. In the *Map* application, attaching a *WallToken* to a DragMag locks it. This is useful to, *e.g.*, prevent any other user from interacting with a DragMag or simply acts as a salient cue that facilitates later access to it. As mentioned above, in the *Art* application, users can also pin a *WallToken* to create a source of water drops.
- ***WallTokens* as identifiers**. In the *Picture* application, each *WallToken* is associated with a distinct selection. Users can add or remove a picture from a selection by tapping it with the associated *WallToken*. Performing a zigzag sliding gesture with a *WallToken* clears the associated selection. As a *WallToken* is uniquely identified, multiple selections do not conflict with each other, making it easy to manage several selections concurrently. In the *Map* application, users perform a tap with a *WallToken* on a DragMag in order to pair them. Once paired, the *WallToken* acts as a proxy to the DragMag, allowing users to manipulate it from anywhere on the wall.
- ***WallTokens* as collaborative tools**. In the *Picture* application, *WallTokens* can be used to implement a range of multi-user contexts. In a competitive context, each user can have their own set of tokens in order to compare individual performance. In a collaborative context, users can work on different regions towards a shared goal, exchanging tokens if needed. As they are uniquely identified, *WallTokens* also enable concurrent *pick-and-drop* operations from multiple users. A *WallToken* can even be duplicated to enable multi-user pick-and-drop where one user picks a picture and the other drops it [29].

We chose this specific set of demo applications as they highlight the unique advantages of large vertical displays: in the *Picture* application, users can step away from the wall in order to view many pictures at once; in the *Map* application, several users can concurrently work with a map displayed in its preferred orientation; and in the *Art* application, the performer can share their creative space with a potentially large audience.

6 LIMITATIONS

Expressive power. In our study, *WallTokens* had advantages over touch gestures in terms of efficiency and comfort. But a single *WallToken* has three degrees of freedom only (2D translation and rotation), while touch gestures can control more degrees of freedom (*e.g.*, pinch to scale). When more degrees of freedom are needed, a combination of touch gestures and tangibles could be interesting (*e.g.*, holding a finger still, close to the token, to switch between rotate and scale modes, or sliding a finger towards or away from the token to control scale). This could avoid users experiencing difficulties when performing multi-touch gestures for the integral control of several dimensions, when they want to act on one dimension without affecting the others [35]. We could also think of more elaborate token designs with *e.g.*, an adjustable cursor as in [43], to increase the number of degrees of freedom that a token can control.

Passive tokens. Our *WallTokens* are fully passive so as to make them compatible with any active display technology. Their mechanism relies on basic supplies only (suction cup and spring). As people are familiar with such supplies, we expect *WallTokens* to afford their manipulations. One first limitation of suction cups is that they work only on smooth surfaces. But this should not be a major issue with most display technologies, as these generally emit light and thus have flat, smooth surfaces to avoid any diffraction effect. A second limitation is that *WallTokens* require an additional action with the push-handle in comparison with approaches based on magnets on a projection-based whiteboard [26]. However, this extra action, which we estimate to take less than 1s in our first experiment, makes *WallTokens* harmless to any active screen technology as opposed to magnets that would damage electronics in screens.

7 CONCLUSION AND FUTURE WORK

We contribute *WallTokens*, which enable tangible user interfaces on vertical displays. *WallTokens*' low-cost design combined with our development framework make it possible to prototype applications that involve multiple tokens. Not only can users manipulate those tokens on the wall, they can also attach them to, and detach them from it at will. This opens up a design space for applications based on multi-token input that could only run on horizontal surfaces until now (*e.g.*, [22, 23]). As users do not have to keep on holding tokens, our contribution also makes tangible input easy to combine with other input channels such as finger touch or pointing devices on vertical displays. Future work includes the design of such interaction techniques that actually combine tangibles with other modalities. We also plan to refine the design of *WallTokens* to track their position and orientation when users hold them in the air, as recently proposed with the concept of *Off-Surface Tangibles* [7], in order to make it possible for users to rely on *WallTokens* for distant interactions with the wall.

ACKNOWLEDGMENTS

We wish to thank Emmanuel Pietriga and all ILDA members for their feedback, Romain Di Vozzo and Jonah Ross-Marrs for their support at the Digiscope Fablab, and Olivier Gladin, Raphaël James and Eugenie Brasier for their help. This research was partially supported by EquipEx Digiscope (EquipEx ANR-10-EQPX-26-01 operated by ANR as part of the program "Investissements d'Avenir").

SUPPLEMENTAL MATERIAL

Details on *WallTokens* fabrication (*e.g.*, SDF files for the different components), experimental data, and the source code of the development framework are available both as supplemental material and online at https://walltokens.lisn.upsaclay.fr.

REFERENCES

[1] Christopher Andrews, Alex Endert, Beth Yost, and Chris North. 2011. Information Visualization on Large, High-Resolution Displays: Issues, Challenges, and Opportunities. *Information Visualization* 10, 4 (Oct. 2011), 341–355. https://doi.org/10.1177/1473871611415997

[2] Alissa N. Antle, Alyssa F. Wise, Amanda Hall, Saba Nowroozi, Perry Tan, Jillian Warren, Rachael Eckersley, and Michelle Fan. 2013. Youtopia: A Collaborative, Tangible, Multi-Touch, Sustainability Learning Activity. In *Proceedings of the 12th International Conference on Interaction Design and Children (IDC '13)*. ACM, 565–568. https://doi.org/10.1145/2485760.2485866

[3] Michel Beaudouin-Lafon, Olivier Chapuis, James Eagan, Tony Gjerlufsen, Stéphane Huot, Clemens Klokmose, Wendy Mackay, Mathieu Nancel, Emmanuel Pietriga, Clément Pillias, Romain Primet, and Julie Wagner. 2012. Multisurface Interaction in the WILD Room. *IEEE Computer* 45, 4 (April 2012), 48–56. https://doi.org/10.1109/MC.2012.110

[4] Andrea Bianchi and Ian Oakley. 2015. MagnID: Tracking Multiple Magnetic Tokens. In *Proceedings of the 9th International Conference on Tangible, Embedded, and Embodied Interaction (TEI '15)*. ACM, 61–68. https://doi.org/10.1145/2677199.2680582

[5] Liwei Chan, Stefanie Müller, Anne Roudaut, and Patrick Baudisch. 2012. CapStones and ZebraWidgets: Sensing Stacks of Building Blocks, Dials and Sliders on Capacitive Touch Screens. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, 2189–2192. https://doi.org/10.1145/2207676.2208371

[6] Olivier Chapuis, Anastasia Bezerianos, and Stelios Frantzeskakis. 2014. Smarties: An Input System for Wall Display Development. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems (CHI '14)*. ACM, 2763–2772. https://doi.org/10.1145/2556288.2556956

[7] Christian Cherek, David Asselborn, Simon Voelker, and Jan Borchers. 2019. Off-Surface Tangibles: Exploring the Design Space of Midair Tangible Interaction. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems (CHI EA '19)*. ACM, Article LBW2115, 6 pages. https://doi.org/10.1145/3290607.3312966

[8] Christian Cherek, Anke Brocker, Simon Voelker, and Jan Borchers. 2018. Tangible Awareness: How Tangibles on Tabletops Influence Awareness of Each Other's Actions. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, Article 298, 7 pages. https://doi.org/10.1145/3173574.3173872

[9] Christian Cherek, Simon Voelker, Jan Thar, Rene Linden, Florian Busch, and Jan Borchers. 2015. PERCs Demo: Persistently Trackable Tangibles on Capacitive Multi-Touch Displays. In *Proceedings of the 2015 International Conference on Interactive Tabletops and Surfaces (ITS '15)*. ACM, 389–392. https://doi.org/10.1145/2817721.2823474

[10] Paul Dourish and Victoria Bellotti. 1992. Awareness and Coordination in Shared Workspaces. In *Proceedings of the 1992 ACM Conference on Computer-Supported Cooperative Work (CSCW '92)*. ACM, 107–114. https://doi.org/10.1145/143457.143468

[11] John Dudley, Hrovje Benko, Daniel Wigdor, and Per Ola Kristensson. 2019. Performance Envelopes of Virtual Keyboard Text Input Strategies in Virtual Reality. In *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR '19)*. IEEE, 289–300. https://doi.org/10.1109/ISMAR.2019.00027

[12] Min Fan, Alissa N. Antle, Carman Neustaedter, and Alyssa F. Wise. 2014. Exploring How a Co-Dependent Tangible Tool Design Supports Collaboration in a Tabletop Activity. In *Proceedings of the 18th International Conference on Supporting Group Work (GROUP '14)*. ACM, 81–90. https://doi.org/10.1145/2660398.2660402

[13] George W. Fitzmaurice and William Buxton. 1997. An Empirical Evaluation of Graspable User Interfaces: Towards Specialized, Space-Multiplexed Input. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI '97)*. ACM, 43–50. https://doi.org/10.1145/258549.258578

[14] Carl Gutwin and Saul Greenberg. 2004. The importance of awareness for team cognition in distributed collaboration. In *Team cognition: Understanding the factors that drive process and performance*. APA, 177–201. https://doi.org/10.1037/10690-009

[15] Mark Hancock, Otmar Hilliges, Christopher Collins, Dominikus Baur, and Sheelagh Carpendale. 2009. Exploring Tangible and Direct Touch Interfaces for Manipulating 2D and 3D Information on a Digital Table. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '09)*. ACM, 77–84. https://doi.org/10.1145/1731903.1731921

[16] Björn Hartmann, Meredith Ringel Morris, Hrvoje Benko, and Andrew D. Wilson. 2010. Pictionaire: Supporting Collaborative Design Work by Integrating Physical and Digital Artifacts. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work (CSCW '10)*. ACM, 421–424. https://doi.org/10.1145/1718918.1718989

[17] Eve Hoggan, John Williamson, Antti Oulasvirta, Miguel Nacenta, Per Ola Kristensson, and Anu Lehtiö. 2013. Multi-Touch Rotation Gestures: Performance and Ergonomics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, 3047–3050. https://doi.org/10.1145/2470654.2481423

[18] Eva Hornecker and Jacob Buur. 2006. Getting a Grip on Tangible Interaction: A Framework on Physical Space and Social Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. ACM, 437–446. https://doi.org/10.1145/1124772.1124838

[19] Robert J. K. Jacob, Linda E. Sibert, Daniel C. McFarlane, and M. Preston Mullen. 1994. Integrality and Separability of Input Devices. *ACM Trans. Comput.-Hum. Interact.* 1, 1 (March 1994), 3–26. https://doi.org/10.1145/174630.174631

[20] Mikkel R. Jakobsen and Kasper Hornbæk. 2014. Up Close and Personal: Collaborative Work on a High-resolution Multitouch Wall Display. *ACM Transactions on Computer-Human Interaction* 21, 2, Article 11 (Feb. 2014), 34 pages. https://doi.org/10.1145/2576099

[21] Yvonne Jansen, Pierre Dragicevic, and Jean-Daniel Fekete. 2012. Tangible Remote Controllers for Wall-size Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, 2865–2874. https://doi.org/10.1145/2207676.2208691

[22] Hans-Christian Jetter, Jens Gerken, Michael Zöllner, Harald Reiterer, and Natasa Milic-Frayling. 2011. Materializing the Query with Facet-streams: A Hybrid Surface for Collaborative Search on Tabletops. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, 3013–3022. https://doi.org/10.1145/1978942.1979390

[23] Sergi Jordà, Günter Geiger, Marcos Alonso, and Martin Kaltenbrunner. 2007. The reacTable: Exploring the Synergy Between Live Music Performance and Tabletop Tangible Interfaces. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction (TEI '07)*. ACM, 139–146. https://doi.org/10.1145/1226969.1226998

[24] Martin Kaltenbrunner, Till Bovermann, Ross Bencina, and Enrico Costanza. 2005. TUIO - A Protocol for Table Based Tangible User Interfaces. In *Proceedings of the 6th International Workshop on Gesture in Human-Computer Interaction and Simulation (GW '05)*. 5 pages.

[25] Sven Kratz, Tilo Westermann, Michael Rohs, and Georg Essl. 2011. CapWidgets: Tangible Widgets Versus Multi-touch Controls on Mobile Devices. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems (CHI EA '11)*. ACM, 1351–1356. https://doi.org/10.1145/1979742.1979773

[26] Jakob Leitner and Michael Haller. 2011. Geckos: Combining Magnets and Pressure Images to Enable New Tangible-Object Design and Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, 2985–2994. https://doi.org/10.1145/1978942.1979385

[27] Rong-Hao Liang, Kai-Yin Cheng, Liwei Chan, Chuan-Xhyuan Peng, Mike Y. Chen, Rung-Huei Liang, De-Nian Yang, and Bing-Yu Chen. 2013. GaussBits: Magnetic Tangible Bits for Portable and Occlusion-free Near-surface Interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, 1391–1400. https://doi.org/10.1145/2470654.2466185

[28] Rong-Hao Liang, Han-Chih Kuo, Liwei Chan, De-Nian Yang, and Bing-Yu Chen. 2014. GaussStones: Shielded Magnetic Tangibles for Multi-token Interactions on Portable Displays. In *Proceedings of the 27th ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, 365–372. https://doi.org/10.1145/2642918.2647384

[29] Can Liu, Olivier Chapuis, Michel Beaudouin-Lafon, and Eric Lecolinet. 2016. Shared Interaction on a Wall-Sized Display in a Data Manipulation Task. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, 2075–2086. https://doi.org/10.1145/2858036.2858039

[30] Can Liu, Olivier Chapuis, Michel Beaudouin-Lafon, and Eric Lecolinet. 2017. CoReach: Cooperative Gestures for Data Manipulation on Wall-Sized Displays. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, 6730–6741. https://doi.org/10.1145/3025453.3025594

[31] Can Liu, Olivier Chapuis, Michel Beaudouin-Lafon, Eric Lecolinet, and Wendy E. Mackay. 2014. Effects of Display Size and Navigation Type on a Classification Task. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, 4147–4156. https://doi.org/10.1145/2556288.2557020

[32] Maurice R. Masliah and Paul Milgram. 2000. Measuring the Allocation of Control in a 6 Degree-of-Freedom Docking Experiment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '00)*. ACM, 25–32. https://doi.org/10.1145/332040.332403

[33] Rafael Morales González, Caroline Appert, Gilles Bailly, and Emmanuel Pietriga. 2016. TouchTokens: Guiding Touch Patterns with Passive Tokens. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, 4189–4202. https://doi.org/10.1145/2858036.2858041

[34] M. R. Morris, A. J. B. Brush, and B. R. Meyers. 2008. A field study of knowledge workers' use of interactive horizontal displays. In *3rd International Workshop on Horizontal Interactive Human Computer Systems (Tabletop '08)*. IEEE, 105–112. https://doi.org/10.1109/TABLETOP.2008.4660192

[35] Miguel A. Nacenta, Patrick Baudisch, Hrvoje Benko, and Andy Wilson. 2009. Separability of Spatial Manipulations in Multi-Touch Interfaces. In *Proceedings of Graphics Interface 2009* (Kelowna, British Columbia, Canada) *(GI '09)*. Canadian Information Processing Society, CAN, 175–182.

[36] Mathieu Nancel, Julie Wagner, Emmanuel Pietriga, Olivier Chapuis, and Wendy Mackay. 2011. Mid-Air Pan-and-Zoom on Wall-Sized Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, 177–186. https://doi.org/10.1145/1978942.1978969

[37] Halla B. Olafsdottir, Theophanis Tsandilas, and Caroline Appert. 2014. Prospective Motor Control on Tabletops: Planning Grasp for Multitouch Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, 2893–2902. https://doi.org/10.1145/2556288.2557029

[38] Brianna Potvin, Colin Swindells, Melanie Tory, and Margaret-Anne Storey. 2012. Comparing Horizontal and Vertical Surfaces for a Collaborative Design Task. *Adv. in Hum.-Comp. Int.* 2012, Article 6 (Jan. 2012), 1 pages. https://doi.org/10.1155/2012/137686

[39] Fateme Rajabiyazdi, Jagoda Walny, Carrie Mah, John Brosz, and Sheelagh Carpendale. 2015. Understanding Researchers' Use of a Large, High-Resolution Display Across Disciplines. In *Proceedings of the International Conference on Interactive Tabletops & Surfaces (ITS '15)*. ACM, 107–116. https://doi.org/10.1145/2817721.2817735

[40] Jun Rekimoto. 1997. Pick-and-drop: A Direct Manipulation Technique for Multiple Computer Environments. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology (UIST '97)*. ACM, 31–39. https://doi.org/10.1145/263407.263505

[41] Yvonne Rogers and Siân Lindley. 2004. Collaborating around vertical and horizontal large interactive displays: which way is best? *Interacting with Computers* 16, 6 (2004), 1133 – 1152. https://doi.org/10.1016/j.intcom.2004.07.008

[42] Bertrand Schneider, Patrick Jermann, Guillaume Zufferey, and Pierre Dillenbourg. 2011. Benefits of a Tangible Interface for Collaborative Learning and Interaction. *IEEE Trans. Learn. Technol.* 4, 3 (July 2011), 222–232. https://doi.org/10.1109/TLT.2010.36

[43] Philip Tuddenham, David Kirk, and Shahram Izadi. 2010. Graspables Revisited: Multi-Touch vs. Tangible Input for Tabletop Displays in Acquisition and Manipulation Tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, 2223–2232. https://doi.org/10.1145/1753326.1753662

[44] Brygg Ullmer, Hiroshi Ishii, and Robert J. K. Jacob. 2005. Token+constraint Systems for Tangible Interaction with Digital Information. *ACM Trans. Comput.-Hum. Interact.* 12, 1 (March 2005), 81–118. https://doi.org/10.1145/1057237.1057242

[45] Simon Voelker, Kjell Ivar undefinedvergård, Chat Wacharamanotham, and Jan Borchers. 2015. Knobology Revisited: A Comparison of User Performance between Tangible and Virtual Rotary Knobs. In *Proceedings of the 2015 International Conference on Interactive Tabletops and Surfaces (ITS '15)*. ACM, 35–38. https://doi.org/10.1145/2817721.2817725

[46] Ulrich von Zadow, Daniel Bösel, Duc Dung Dam, Anke Lehmann, Patrick Reipschläger, and Raimund Dachselt. 2016. Miners: Communication and Awareness in Collaborative Gaming at an Interactive Display Wall. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces (ISS '16)*. ACM, 235–240. https://doi.org/10.1145/2992154.2992174

[47] Ulrich von Zadow, Wolfgang Büschel, Ricardo Langner, and Raimund Dachselt. 2014. SleeD: Using a Sleeve Display to Interact with Touch-sensitive Display Walls. In *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces (ITS '14)*. ACM, 129–138. https://doi.org/10.1145/2669485.2669507

[48] Manuela Waldner, Jörg Hauber, Jürgen Zauner, Michael Haller, and Mark Billinghurst. 2006. Tangible Tiles: Design and Evaluation of a Tangible User Interface in a Collaborative Tabletop Setup. In *Proceedings of the 18th Australia Conference on Computer-Human Interaction: Design: Activities, Artefacts and Environments (OZCHI '06)*. ACM, 151–158. https://doi.org/10.1145/1228175.1228203

[49] Colin Ware and Marlon Lewis. 1995. The DragMag Image Magnifier. In *Conference Companion on Human Factors in Computing Systems (CHI '95)*. ACM, 407–408. https://doi.org/10.1145/223355.223749

[50] Malte Weiss, Julie Wagner, Yvonne Jansen, Roger Jennings, Ramsin Khoshabeh, James D. Hollan, and Jan Borchers. 2009. SLAP Widgets: Bridging the Gap Between Virtual and Physical Controls on Tabletops. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, 481–490. https://doi.org/10.1145/1518701.1518779

[51] Malte Weiss, Julie Wagner, Roger Jennings, Yvonne Jansen, Ramsin Khoshabeh, James D. Hollan, and Jan Borchers. 2009. SLAPbook: Tangible Widgets on Multi-Touch Tables in Groupware Environments. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (TEI '09)*. ACM, 297–300. https://doi.org/10.1145/1517664.1517725

[52] Jian Zhao, R. William Soukoreff, and Ravin Balakrishnan. 2015. Exploring and Modeling Unimanual Object Manipulation on Multi-Touch Displays. *Int. J. Hum.-Comput. Stud.* 78, C (June 2015), 68–80. https://doi.org/10.1016/j.ijhcs.2015.02.011